Breaking the Deadlock: Simultaneously Discovering Attribute Matching and Cluster Matching with Multi-Objective Simulated Annealing

Haishan Liu and Dejing Dou

Computer and Information Science Department,
University of Oregon, Eugene, USA

Abstract. In this paper, we present a data mining approach to challenges in the matching and integration of heterogeneous datasets. In particular, we propose solutions to two problems that arise in combining information from different results of scientific research. The first problem, *attribute matching*, involves discovery of correspondences among distinct numeric-typed summary features ("attributes") that are used to characterize datasets that have been collected and analyzed in different research labs. The second problem, *cluster matching*, involves discovery of matchings between patterns across datasets. We treat both of these problems together as a multi-objective optimization problem. A multi-objective simulated annealing algorithm is described to find the optimal solution. The utility of this approach is demonstrated in a series of experiments using synthetic and realistic datasets that are designed to simulate heterogeneous data from different sources.

Keywords: Multi-Objective Optimization, Cluster Matching, Attribute Matching, Simulated Annealing.

1 Introduction

The presence of heterogeneity among schemas and ontologies supporting vast amount of informational sources leads to one of the most important and toughest problems, that is, the semantic integration of heterogeneous data sources to facilitate interoperability and reuse of the information. The difficulty is especially pronounced in many scientific domains where massive amount of data are produced independently and thus each having their own data vocabulary. While manual integration is time-consuming and requires expensive specialized human capital, the development of automatic approaches becomes imminent to aid inter-institute collaboration. One purpose of the present paper is to suggest a method for solving a specific kind of ontology/schema matching problem under some severe constraints that can cause traditional methods to be ineffective. The constraints that we deal with are, namely, 1) little-to-no string-based or linguistic similarity between terminologies, and 2) all numeric typed data instances. This phenomenon is commonly seen in integrating scientific datasets

R. Meersman, T. Dillon, and P. Herrero (Eds.): OTM 2011, Part II, LNCS 7045, pp. 698–715, 2011.

which involves discovery of correspondences among distinct numeric-typed summary features ("attributes") that are used to characterize datasets that have been collected and analyzed in different research labs. We call this the *attribute matching* problem.

Another challenging task given multiple data sources is to carry out meaningful meta-analysis that combines results of several studies on different datasets to address a set of related research hypotheses. Finding correspondences among distinct patterns that are observed in different scientific datasets is an example of meta-analysis. Supposing the patterns are derived by clustering analysis, this problem can be addressed by the application of cluster comparison (or cluster matching) techniques. Clustering is an unsupervised data mining task widely used to discover patterns and relationships in a variety of fields. The clustering result provides a pattern characterization from a data-driven perspective. If similar results are obtained across multiple datasets, this leads in turn to a revision and refinement of existing domain knowledge, which is a central goal of meta-analysis. However, there are noticeably few cluster comparison methods that are able to compare two clusterings derived from different datasets. The difficulty for the comparison is further exacerbated by the fact that the datasets may be described by attributes from heterogeneous ontologies or schemas. Even those methods that are able to measure clustering similarity across different datasets (e.g., the ADCO [1] method) have to assume the homogeneous meta-data.

Given this situation, in order to carry out cluster comparison for meta-analysis, researchers often need to perform ontology or schema matching first in order to mitigate the meta-data gap. In previous work [11], we examine a practical attribute matching problem on neuroscience data where schema elements from one dataset share no lexical similarity with those from the other. Moreover, structural similarity is also limited. One can only resort to instance-based (extensional) methods. However, since all attributes are numerical, information clues available to an instance-level matcher is very restricted. Traditional instance-based matchers typically make use of constraint-based characterization, such as numerical value ranges and averages to determine correspondences. However, this is often too rough in the case of all-numerical dataset. Two attributes may have similar ranges and averages but totally different internal value distributions (an example is shown in Section 4.1). Given this, we propose to represent the attribute value distribution at a finer granularity by partitioning the values into groups. To do this, clustering is performed, and resulting clusters are then aligned across two datasets (assuming that the same pattern exists in both datasets). In this way, each attribute can be characterized by, instead of a single value, a vector of per-cluster statistical quantities (i.e., the *segmented statistical characterization*). A distance function can then be applied based on this representation. Table 1(A) shows an example distance table on the cross join of two sets of attributes. To discover attribute matching from this table can be reduced to solving a minimum assignment problem (assuming matching is bijective), which is a classical combinatory optimization problem that has a polynomial solution using the Hungarian Method [8].

Unfortunately, however, the above solution requires us to be able to align clusters across datasets, which is a difficult problem in its own right. If fully automated, as mentioned above, methods such as ADCO adopt a so called *density profile* [1] representation of clusters that requires homogeneous meta-data or a priori knowledge about the attribute matching in heterogeneous scenarios. Then the cluster matching can be carried out in a similar manner to the attribute matching by casting to the assignment problem (see Table 1(B), for example). This leads to a circular causality, or a deadlock, between the attribute matching (under the segmented statistical characterization) and cluster matching (under the density profile representation) problems—none of them can be solved automatically without the other one being solved first.

Table 1. Example distance matrices between (A) two sets of attributes and (B) two sets of clusters, respectively

	a'_1	a'_2	$\cdots$	a'_m
a_1	$d_{11'}$	$d_{12'}$	$\cdots$	$d_{1m'}$
a_2	$d_{21'}$	$d_{22'}$		$d_{2m'}$
$\vdots$			$\ddots$	
a_n	$d_{n1'}$	$d_{n2'}$		$d_{nm'}$

(A)

	c'_1	c'_2	$\cdots$	c'_n
c_1	$d_{11'}$	$d_{12'}$	$\cdots$	$d_{1n'}$
c_2	$d_{21'}$	$d_{22'}$		$d_{2n'}$
$\vdots$			$\ddots$	
c_n	$d_{n1'}$	$d_{n2'}$		$d_{nn'}$

(B)

To solve this difficulty, in the present paper, viewing the two matching problems as combinatorial optimization problems with distinct yet interrelated objective functions, we propose a novel approach using a multi-objective simulated annealing (MOSA) to discover attribute matching and cluster matching simultaneously. The objectives in the optimization are to minimize distances of attribute matching and cluster matching respectively.

The rest of this paper is organized as follows. We review the basics of multi-objective optimization and describes the relationship between various components of the proposed method and existing methods in Section 2. We present detailed description of our method for simultaneously discovering attribute matching and cluster matching in Section 3. We report experimental results in Section 4 and conclude the paper in Section 5.

2 Background and Related Work

2.1 The Multiobjective Optimization Problem and Pareto-Optimality

Multi-objective optimization problem (also called multi-criteria, multi-performance or vector optimization) can be defined mathematically as to find the vector $X = [x_1, x_2, \ldots, x_k]^T$ which satisfies the following m inequality constraints and l equality constraints:

$$g_i(X) \geq 0, i = 1, 2, \ldots, m$$
$$h_i(X) = 0, i = 1, 2, \ldots, l$$

and optimize the objective function vector

$$F(X) = [f_1(X), f_2(X), \ldots, f_N(X)]^T$$

where $X = [x_1, x_2, \ldots, x_k]^T$ is called the decision variable vector.

Real-life problems require simultaneous optimization of several incommensurable and often conflicting objectives. Usually, there is no single optimal solution, but there is a set of alternative solutions. These solutions are optimal in the sense that no other solutions in the search space are superior to each other when all the objectives are considered [16]. They are known as Pareto-optimal solutions. To define the concept of Pareto optimality, we take the example of a minimization problem with two decision vectors $a, b \in X$. Vector a is said to dominate b if

$$\forall i = \{1, 2, \ldots, N\} \ : \ f_i(a) \leq f_i(b)$$
$$and$$
$$\exists j = \{1, 2, \ldots, N\} \ : \ f_j(a) < f_j(b)$$

When the objectives associated with any pair of non-dominated solutions are compared, it is found that each solution is superior with respect to at least one objective. The set of non-dominated solutions to a multi-objective optimization problem is known as the Pareto-optimal set (Pareto front) [17].

2.2 Simulated Annealing in Multi-Objective Optimization

Simulated annealing (SA) is based on an analogy of thermodynamics with the way metals cool and anneal. It has been proved to be a compact and robust technique, which provides excellent solutions to single and multiple objective optimization problems with a substantial reduction in computation time. It is a method to obtain an optimal solution of a single objective optimization problem and to obtain a Pareto set of solutions for a multi-objective optimization problem. Simulated Annealing was started as a method or tool for solving single objective combinatorial problems, these days it has been applied to solve single as well as multiple objective optimization problems in various fields. A comprehensive survey can be found in [16].

2.3 The Schema Matching Problem

Our study of matching alternative attribute sets is closely related to the schema matching problem. According to the type of instance value, various instance-based approaches have been developed in previous research. For example, for textual attributes, a linguistic characterization based on information retrieval techniques can be applied [12]; for nominal attributes, evaluation of the degree of overlap of instance values is a preferred approach. Larson et al. [9] and Sheth et al. [14] discussed how relationships and entity sets could be integrated primarily based on their domain relationships. Similarity of partially overlapped instance set can be also calculated based on measures such as Hamming distance

and Jaccard coefficient; for numeric attributes, most methods use aggregated statistics to characterize the attributes, e.g., 'SSN' and 'PhonNo' can be distinguished based on their respective patterns [12]. Hybrid systems that combine several approaches to determine matching often achieve better performance. For example, SemInt [10] is a comprehensive matching prototype exploiting up to 15 constraint-based and 5 content-based matching criteria. The LSD (Learning Source Descriptions) [4] system uses several instance-level matchers (learners) that are trained during a preprocessing step. The iMAP [2] system uses multiple basic matchers, called searches, e.g., text, numeric, category, unit conversion, each of which addresses a particular subset of the match space.

Due to the nature of many scientific datasets, we face several unique challenges. First, the data under study are semi-structured, thus invalidating those matching methods that presume a complete, known-in-advance schematic structure. In addition, totally different labels (usually acronyms or pseudowords) are widely adopted for the same or similar metrics, rendering lexical similarity-based methods unsuitable. Moreover, an important limitation of previous instance-based matching methods is their inability to handle numerical instances appropriately in certain domain applications. They use statistical characterization extracted from the numerical instances, such as range, mean and standard deviation, to determine match. However such information is too rough to capture patterns in data that are crucial in determining the correspondence.

2.4 The Cluster Matching Problem

The cluster matching (cluster comparison) problem is related to the cluster validity problem, especially the technique of external/relative indexing that aims at comparing two different clustering results. Popular methods in this field, including the Rand index [13], Jaccard index [7], normalized mutual information [5], etc., are mostly based on examining membership of points to clusters. However, the basis of these methods is the comparison of different clustering schema on the same dataset.

By contrast, in the present case we are aiming to match clusters across datasets that contain non-overlapping observations. Thus, membership-based cluster validity criteria are unsuitable. A recent clustering similarity index known as ADCO (Attribute Distribution Clustering Orthogonality) proposed by Bae et al. [1] can match clusterings from non-overlapping datasets. The ADCO measure determines the similarity between two clusterings based on their *density profiles*, which incorporate distribution information of data points along each attribute. The density profile representation of clusters are defined as follows.

Density Profile: To represent clusters using density profiles, the attribute's range in each cluster is first discretized into a number of bins, and the similarity between two clusters corresponds to the number of points of each cluster falling within these bins. The formal definition for this number of points is the *density* of an attribute-bin region for cluster c_k in clustering C, denoted as $dens_C(k, i, j)$. It refers to the number of points in the region (i, j)—the j-th bin of the i-th

attribute—that belongs to the cluster c_k of clustering C. For example, for clustering C in Fig. 1, $dens_C(1,1,1) = 8$, because there are 8 data points in region $(1,1)$—the first bin of the first attribute x—that belongs to the first cluster c_1.

The density profile vector V_C for a clustering C is formally defined as an ordered tuple:

$$V_C = \Big[dens_C(1,1,1),\ dens_C(1,1,2),\ldots,\ dens_C(1,1,Q),\ dens_C(1,2,1),$$
$$\ldots,\ dens_C(1,M,Q),\ dens_C(2,1,1),\ldots,\ dens_C(N,M,Q)\Big], \quad (1)$$

where Q is the number of bins in each of the M attributes, and K is the number of clusters in C.

The ADCO measure: After the density profile vectors of two clusterings C and C' are obtained, the degree of similarity between C and C' can be determined by calculating the dot product of the density profile vectors:

$$sim(C, C') = V_C \cdot V_{C'}\,.$$

Given a permutation π under which the similarity function $sim(C,\ \pi(C'))$ is maximized, an ADCO measure is calculated using a normalization factor (NF) corresponding to the maximum achievable similarity of the clusterings: $NF(C,C') = max\big[sim(C,C),\ sim(C',C')\big]$. The $ADCO(C,C')$ measure is defined as follows:

$$ADCO(C,C') = \frac{sim(C,C')}{NF(C,C')}\,.$$

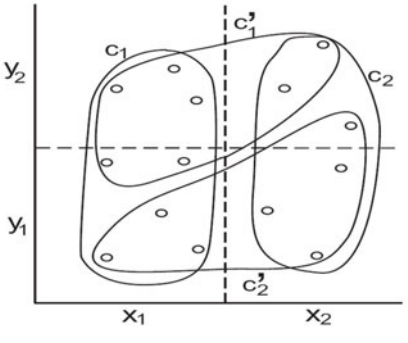

Fig. 1. Two clusterings $C = \{c_1, c_2\}$ and $C' = \{c'_1, c'_2\}$. Two attributes X (attribute 1) and Y (attribute 2) are discretized into 2 bins each. See [1] for details.

3 Method

3.1 The Multi-Objective Simulated Annealing Framework

To solve the dual matching problems, we adopt a strategy of multi-objective simulated annealing described in [15], in which the acceptance criterion in the simulated annealing process is established based on the idea of Pareto-domination based fitness. Fitness of a solution is defined as one plus the number of dominating solutions in Pareto-optimal set. The larger the value of fitness, the worse is the solution. Initially, fitness difference between the current and the generated solution is less and the temperature is high so any move is accepted due to both of them. This gives a way to explore the full solution space. As the number of iterations increases, temperature decreases and fitness difference between the current and generated solutions may increase. Both of them make the acceptance move more selective and it results in a well-diversified solution in true Pareto-optimal solutions. Details of our adaptation of the above multi-objective simulated annealing framework is outlined in Algorithm 1.

Algorithm 1. Multi-Objective Simulated Annealing

Input: Empty Pareto-optimal set of solutions $\mathbf{\Sigma}$
Input: Empty current decision vector $\mathbf{X} = [x_a, x_c]$
Input: Initial temperature T
  $count = 0$
  while $T > threshold$ **do**
    $initialize(\mathbf{X})$
    Put $\mathbf{X}$ in $\mathbf{\Sigma}$
    $\mathbf{X}' = generate\_solution(\mathbf{X})$
    $S_{\mathbf{X}'} = evaluate\_solution(\mathbf{X}')$
    $\Delta S = S_{\mathbf{X}'} - S_X$
    if $r = rand(0,1) < exp(\frac{-\Delta S}{T})$ **then**
      $\mathbf{X} = \mathbf{X}'$
      $S_X = S_{X'}$
    end if
    //Periodically restart
    if $count == restart\_limit$ **then**
      $\mathbf{X} = select\_random\_from\_Pareto(\mathbf{\Sigma})$
      continue
    end if
    $reduce\_temperature(T)$
  end while

Mathematically, the processes involved in the proposed multi-objective simulated annealing framework can be defined as follows.

$$
\begin{aligned}
X &= [x_a, x_c] \\
F &= [f_a, f_c] \\
P_a([x_a^{(n-1)}, x_c^{(n-1)}]) &= [x_a^{(n)}, x_c^{(n-1)}] \\
P_c([x_a^{(n-1)}, x_c^{(n-1)}]) &= [x_a^{(n-1)}, x_c^{(n)}] \\
G_{c|a}([x_a^{(n)}, x_c^{(n-1)}]) &= [x_a^{(n)}, x_c^{(n)}] \\
G_{a|c}([x_a^{(n-1)}, x_c^{(n)}]) &= [x_a^{(n)}, x_c^{(n)}] \\
G \circ P([x_a^{(n-1)}, x_c^{(n-1)}]) &= [x_a^{(n)}, x_c^{(n)}]
\end{aligned}
$$

X is the decision vector that contains two variables for attribute matching, x_a, and cluster matching, x_c, respectively (details in Section 3.2). F is the objective function vector that contains two criterion functions (f_a and f_c) to evaluate attribute matching and cluster matching decisions (details in Section 3.4). P is the random perturbation function that takes a decision vector in the $(n-1)$th iteration and partially advances it to the nth iteration (we use P_a or P_c to distinguish between the random selections). The partial candidate decision generation function G takes the output of P and fully generate a decision vector for the nth iteration (by advancing the left-out variable in P to its nth iteration). Thus, the compound function $G \circ P$ fulfils the task of generating an nth-iteration candidate decision vector given the $(n-1)$th one (details in Section 3.5).

3.2 Decision Variable

The domains of the decision variables in the matching problems take values on a permutation space. In other word, by formalizing the problem of finding correspondent elements of two sets S and S' of cardinality n as an optimization problem, the solution is completely specified by determining an optimal permutation of $1, \ldots, n$. For instance, for two sets of three elements, their indexes range over $\{0, 1, 2\}$. Applying a permutation $\pi = \{2, 0, 1\} \in S_3$ on S' can be viewed as creating a mapping (bijection) from elements on the new positions of S' to elements on the corresponding positions in S. In this example, the permutation π on S' specifies the following correspondences: $S_0 \leftrightarrow S'_2$, $S_1 \leftrightarrow S'_0$, and $S_2 \leftrightarrow S'_1$.

Formally, let P_n ($n \in \mathbb{N}$) be the symmetric group of all permutations of the set $\{1, 2, \ldots, n\}$. Given two sets S and S' with the same cardinality of n, performing identity permutation on one set and an arbitrary permutation $\pi \in S_n$ on the other specifies a matching (or mathematically speaking, mapping) between the two sets. In the multi-objective optimization formalism for solving attribute matching and cluster matching problems, the decision vector has two variables: $X = [x_a, x_c]$. If we have M attributes and N clusters to match respectively, then $x_a \in P_M$ and $x_c \in P_N$.

3.3 Data Representation

The central objects of interest in our study, namely, the numeric-typed attributes and clusters, need to be represented in ways that meaningful quantities can

be defined to measure the "goodness" of a matching decision. To this end, we propose to use the *segmented statistical characterization* to represent attributes, and the *density profiles* to represent clusters. Details of these representations are described below.

Representation of Attributes: Numeric-typed attributes can be represented by the segmented statistical characterization, in which data instances are first partitioned into groups (e.g., through unsupervised clustering) and then characterized by a vector of indicators, each denoting a statistical characterization of the corresponding group. For example, if values of an attribute A are clustered into n groups, then it can be represented by a vector of segmented statistical characterization as follows:

$$V_A = \Big[\mu_1, \mu_2, \ldots, \mu_n\Big],$$

where we choose the mean value μ_i for cluster i as the statistical indicator in our implementation.

Representation of Clusters: Clusters can be represented using density profiles [1] as described in Section 2. The attribute's range in each cluster is first discretized into a number of bins, and the similarity between two clusters corresponds to the number of points (i.e. *density*) of each cluster falling within these bins. Given this, density profile vector V_C for a clustering C is formally defined as an ordered tuple by Equation 1 and is repeated here:

$$V_C = \Big[dens_C(1,1,1),\ dens_C(1,1,2),\ldots,\ dens_C(1,1,Q),\ dens_C(1,2,1),$$
$$\ldots,\ dens_C(1,M,Q),\ dens_C(2,1,1),\ldots,\ dens_C(N,M,Q)\Big],$$

where Q is the number of bins in each of the M attributes, N is the number of clusters in C, and $dens_C(k,i,j)$ refers to the number of points in the region (i,j)—the j-th bin of the i-th attribute—that belongs to the cluster c_k of clustering C.

3.4 Objective Functions

The objective functions in the attribute matching and cluster matching problems are criteria to evaluate the "goodness" of matchings. We use the sum of pair-wise distances between matched elements (see Figure 1 for example) as the objective function. Given this, to determine the form of objective functions amounts to defining proper pair-wise distance measures for the attribute and cluster matching problems respectively, as detailed in the following.

Distance function between two attributes. The pairwise distance between two attributes are defined as the Euclidean distance between their segmented

statistical characterization vectors, and f_a calculates the sum of pair-wise distances under the attribute matching specified by x_a:

$$f_a(x_a) = \sum_{k=1}^{M} \mathcal{L}\left((V_a)^k,\ (V'_a)^{x_a(k)} \right) = \sum_{k=1}^{M} \sqrt{\sum_{i=1}^{N} \left(\mu_i^k - (\mu')_i^{x_a(k)} \right)^2}, \tag{2}$$

where $x_a \in P_M$.

Distance function between two clusters. The ADCO similarity described in Section 2.4 can be transformed to a distance defined as follows [1]:

$$D_{ADCO}(C, C') = \begin{cases} 2 - ADCO(C, C') & , \ if\ C \neq C'\ (V_C \neq V_{C'}) \\ 0 & , \ otherwise \end{cases} \tag{3}$$

We use D_{ADCO} as the pair-wise distance between two clusters under the density profile representation, and f_c calculates the sum of pair-wise distances under the cluster matching specified by x_c

$$f_c(x_c) = \sum_{k=1}^{N} D_{ADCO}\left((V_c)^k,\ (V'_c)^{x_c(k)} \right) = \sum_{k=1}^{N} \left(2 - \frac{\sum_{i=1}^{M}\sum_{j=1}^{Q} \left(dens(k,i,j) \times dens(x_c(k),i,j) \right)}{\max\left[\sum_{i=1}^{M}\sum_{j=1}^{Q} dens(k,i,j)^2\ ,\ \sum_{i=1}^{M}\sum_{j=1}^{Q} dens(x_c(k),i,j)^2 \right]} \right), \tag{4}$$

where $x_c \in P_N$.

3.5 Generation of New Solution

In each iteration of the simulated annealing process, we randomly generate candidate decision in the neighborhood of the last-iteration decision by applying two consecutive processes, namely, the random perturbation and the partial candidate decision generation, as described below.

Random Perturbation: In each iteration, we select at random one variable (either x_a or x_c) in the decision vector and perturb it by randomly swapping two positions in the selected variable. This advances that variable from $(n-1)$th iteration to nth iteration. Then the following partial candidate generation process is carried out to bring the other variable also to nth iteration.

Partial candidate decision generation
Given $x_c^{(n)}$, derive $x_a^{(n)}$:

$$x_a^n = \arg\min_{\pi} f_a(\pi, x_c^{(n)}) = \arg\min_{\pi} \sum_{k=1}^{M} \mathcal{L}\left((V_a)^k,\ (V_a')^{\pi(k)}\right)$$
$$= \arg\min_{\pi} \sum_{k=1}^{M} \sqrt{\sum_{i=1}^{N} \left(\mu_i^k - (\mu')_{x_c^{(n)}(i)}^{\pi(k)}\right)^2} \tag{5}$$

Given $x_a^{(n)}$, derive $x_c^{(n)}$:

$$x_c^n = \arg\min_{\pi} f_c(\pi, x_a^{(n)}) = \arg\min_{\pi} \sum_{k=1}^{N} D_{ADCO}\left((V_c)^k,\ (V_c')^{\pi(k)}\right)$$

$$= \arg\min_{\pi} \sum_{k=1}^{N} \left(2 - \frac{\sum_{i=1}^{M}\sum_{j=1}^{Q}\left(dens(k,i,j) \times dens(\pi(k), x_a^{(n)}(i), j)\right)}{\max\left[\sum_{i=1}^{M}\sum_{j=1}^{Q} dens(k,i,j)^2\ ,\ \sum_{i=1}^{M}\sum_{j=1}^{Q} dens(\pi(k), x_a^{(n)}(i), j)^2\right]}\right) \tag{6}$$

To calculate π that satisfies equations 5 and 6, rather than iterating through all possible permutations, we can consider the equation as a minimum-cost assignment problem. Table 1(A), for example, illustrates a distance table between two attribute sets A and A'. Matching of the two sets can be considered as an assignment problem where the goal is to find an assignment of elements in $\{A_i\}$ to those in $\{A_i'\}$ that yields the minimum total distance without assigning each A_i more than once. This problem can be efficiently solved by the Hungarian Method in polynomial time of $O(K_{min}^3)$ [8]. It is worth noting that by formulating the problem as the assignment problem, we assume the matching between two sets to be a one-to-one function.

4 Experiment

Because we are interested in understanding the property of the Pareto front obtained by our method, we conducted a series of experiments to highlight tradeoffs of the objectives functions. First, to illustrate the proposed method is indeed capable of determining matching between numeric-typed attributes and clusters, we synthesized a dataset simulating some extreme conditions under which previous methods are ineffective. Also, from the results obtained on the synthetic dataset, we empirically study tradeoffs between the two objective functions. Then, to evaluate the scalability of the method, we carry out a series of tests on a set of data with varied sizes. Finally, encouraged by these results, we applied our methods to actual neuroscience ERP (event-related potentials) data to highlight the applicability of our method to the neuroscience domain.

4.1 Synthetic Dataset

Data Generation: In the synthetic dataset, we generated values for each attribute in such a way that each attribute can be divided into several clusters, and each cluster corresponds to a Gaussian distribution with different mean and standard deviation, but the overall mean and standard deviation of values from all clusters in one attribute are made very close to those in other attributes. For example, Figure 2 illustrates the value distributions of three attributes (a_1, a_2, and a_3) from one dataset and their corresponding counterparts (a'_1, a'_2, and a'_3) from another. It shows that the overall means and standard deviations for these six attributes are almost indistinguishable, and their ranges are similar as well. Previous methods using these whole-attribute-wise quantities as statistical characterization of attributes would have a hard time determining the matchings. However, as mentioned above and illustrated in the figure, the individual distributions underlying clusters in these attributes are distinct and, by using the segmented statistical characterization of attributes, the difference is significant enough to differentiate and identify matchings between attributes.

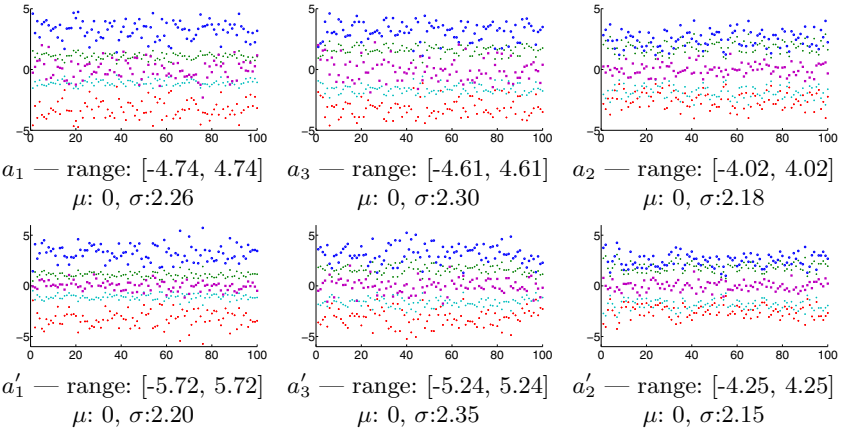

Fig. 2. Scatter plots of data instances from three sample attributes in one synthetic dataset (upper frame) and those of their corresponding attributes from another (lower frame) are illustrated to show their respective value distributions

Results: Figure 3 illustrates the Pareto front obtained from matching two synthetic datasets, each having 20 attributes and 5 clusters. Most notably, the gold standard results for both attribute matching and cluster matching are obtained from the left-most point on the Pareto front. In other words, given the decision variables (X) corresponding to that point, we obtained 100% correct matching results. We further observed that in our subsequent tests on other synthetic datasets with varied number of attributes and clusters, the derived Pareto fronts all contain gold standard result, and the point corresponding to the gold standard can always be found towards the minimum end of f_a. Given this, we propose

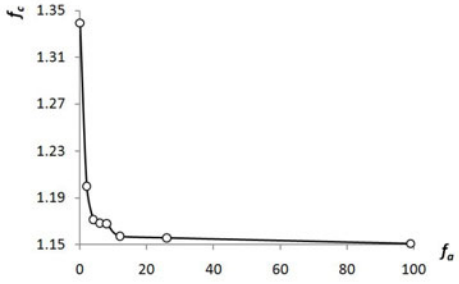

Fig. 3. An example Pareto front obtained from matching two synthetic datasets with 20 attributes and 5 clusters

the following method to reduce the Pareto-optimal set to a single point corresponding to the most favored choice (X^*) in the decision space. The idea is to find the decision with the minimum weighted sum of objective values in the obtained Pareto-optimal set, i.e., $X^* = \arg\min_X \left[\alpha f_a(X) + \beta f_c(X)\right]$, where α and β are weights. We first conducted preliminary experiments to determine the best values for α and β (0.8 and 0.2 respectively) and used them in all subsequent experiments. This method works markedly well on the synthetic datasets. For all the tests described in Table 2, 100% correct results for both attribute and cluster matchings are obtained (hence we omit the precision in the table).

Running Time: We systematically altered the number of attributes and clusters present in the data and conducted a series of tests to show the scalability of the proposed method. The running time under different configurations is reported in Table 2. The time is calculated by averaging over 5 runs of each test (on a 2.53GHz dual-core CPU with 4 gigabytes memory), each run having 1000 iterations in the simulated annealing process. The main computationally expensive part of the annealing process is the generation of new candidate solution phase (function G) in which an assignment problem is solved using the Hungarian method. The complexity of the Hungarian method is cubic and is already the most efficient algorithm for solving the assignment problem (a brute force algorithm has a factorial complexity). Fortunately, rarely is the case that the number of attributes or clusters is large in real-world scenarios where the proposed technique is needed. For reasonable configurations in most practical applications, the computation time is within a tractable range as shown in table 2.

4.2 Neuroscience Dataset

Data Acquisition: To address the problems of attribute and cluster matching in a real-world neuroscience application, we used a set of realistic simulated ERP (event-related potentials) datasets, which were designed to support evaluation of ERP analysis methods [6]. The datasets were specifically designed to simulate heterogeneous data from different groups of subjects under different conditions (via distinct simulated brain activities), as well as distinct measurement methods

Table 2. Running time of the annealing process on synthetic datasets with varied configurations of attribute and cluster sizes. The time is obtained by averaging over results of 5 runs of each test.

# attributes	# clusters	time (sec)
5	20	0.28
20	20	1.81
20	40	7.04
20	60	17.80
40	20	4.66
40	40	11.74
40	60	25.93
60	20	10.95
60	40	20.70
60	60	37.35
100	100	172.23

(spatial and temporal metrics) and distinct patterns (reflecting two different pattern decomposition techniques). Real ERP data arise from superposition of latent scalp-surface electrophysiological patterns, each reflecting the activity of a distinct cortical network that cannot be reconstructed from the scalp-measured data with any certainty. Thus, real ERP data are not appropriate for evaluation of ERP pattern mapping. By contrast, simulated ERP data are derived from known source patterns and therefore provide the necessary gold standard for evaluation of our proposed methods.

The raw data for this study consist of 80 simulated event-related potentials (ERPs), in which each ERP comprises simulated measurement data for a particular subject ($n = 40$). The 40 simulated subjects are randomly divided into two 20-subject groups, SG1 and SG2, each containing 40 ERPs (20 subjects in 2 experimental conditions). Each ERP consists of a superposition of 5 latent varying spatiotemporal patterns. These patterns were extracted from the two datasets, SG1 and SG2, using two techniques: temporal Principal Components Analysis (tPCA) and spatial Independent Components Analysis (sICA), two data decomposition techniques widely used in ERP research [3]. To quantify the spatiotemporal characteristics of the extracted patterns, two alternative metric sets, m1 and m2, were applied to the two tPCA and the two sICA derived datasets. For a complete explanation of these alternative metrics, please see Appendix in [6].

In summary, the simulated ERP data generation process yielded eight test datasets in total, reflecting a 2 (attribute sets) $\times$ 2 (subject groups) $\times$ 2 (decomposition methods) factorial design. Therefore, for each attribute sets there are 4 datasets generated from different combinations of subject groups and decomposition methods, resulting $4 \times 4 = 16$ cases for the studies of attribute matching and cluster matching. The reason to include such variabilities was to test the robustness of our matching method to different sources of heterogeneity across the different datasets. Within all test datasets, 5 major ERP spatiotemporal patterns are present. They are P100, N100, N3, MFN, and P300.

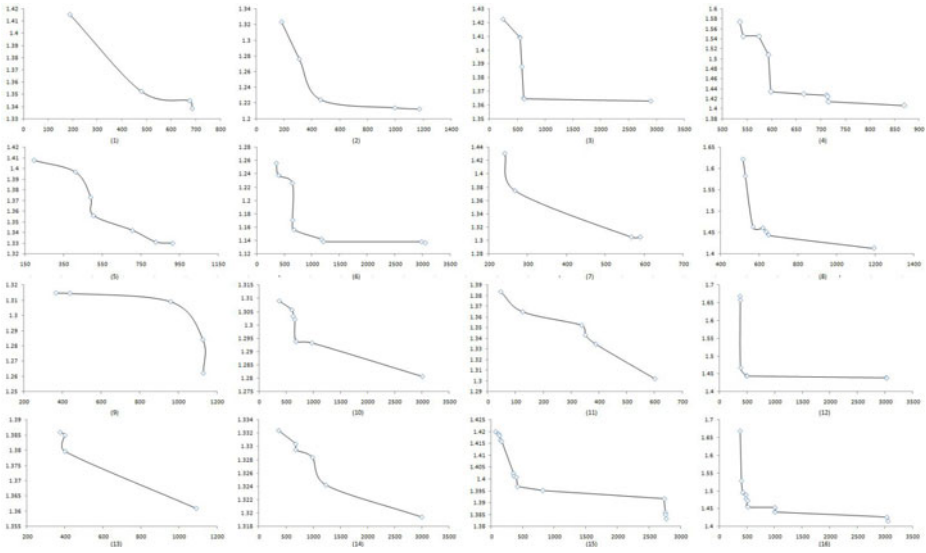

Fig. 4. Pareto fronts obtained from the 16 test cases of the neuroscience dataset

These patterns can be identified in the datasets by clustering analysis. Pretending that the latent patterns underlying discovered clusters are unknown, we hope to match clusters across datasets to recover the fact that the same patterns are present in all datasets.

Results: Figure 4 illustrates the Pareto fronts derived by the proposed method on each of the 16 test cases. We applied the weighted sum method to determine the most favored choice from the Pareto fronts using the parameters (α and β) discovered in the preliminary experiments on synthetic datasets (cf. Section 4.1). The accuracy of attribute matching and cluster matching along with the number of points in the Pareto front are listed in Table 3 (all these results are obtained by taking average from 5 runs for each test case).

It can be observed from the results in Table 3 that more different factors involved in the acquisition of the two datasets for matching can negatively affect the matching performance. For example, in test case 1, the two datasets are drawn from the same subject group (SG1) and preprocessed using the same decomposition method (sICA); whereas in test case 4, the subject groups and decomposition methods are all different, resulting in greater variability and hence the performance is less satisfactory. However, it is worth noting that our method greatly outperforms traditional whole-attribute-based statistic characterization, as is shown in Table 5. In this table we also demonstrate the accuracy of the segmented statistics characterization with expert-labeled patterns, meaning that the data is partitioned and aligned in the most accurate way, which marks the best achievable attribute matching performance. But it is not feasible because, as mentioned in Section 1, manually recognizing patterns (partitioning data) and aligning them across datasets requires

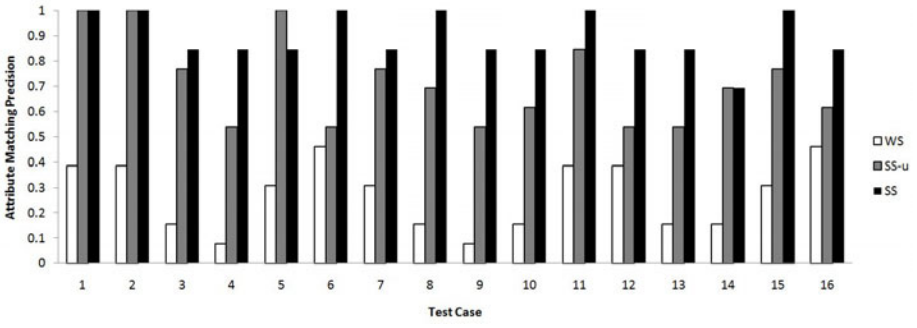

Fig. 5. A comparison of the attribute matching accuracy of three methods on the 16 test cases of the neuroscience dataset. The three methods being compared are matching based on whole-attribute statistics (WS), segmented attribute statistics without knowing a priori cluster matching (SS-u), and segmented attribute statistics with expert-aligned clusterings (SS).

a priori knowledge of attributes in the datasets which is exactly what the problem of attribute matching tries to discover (the circular causality problem). On the other hand, our method does not require human involvement (except the specification of the number of clusters (patterns) present in the data in order to run the clustering analysis) in determining both the attribute matching and cluster matching and is able to achieve close-to-optimal results.

Table 3. Matching performance of the proposed method on the 16 test cases from the neuroscience dataset. The source and target parameter configuration of the data acquisition process of each test case are shown. P_a and P_c denote the accuracy of attribute matching and cluster matching respectively. Σ is the number of points in the obtained Pareto-front. The quantities listed in the table are obtained by averaging over 5 runs of each test.

Test case	Source params	Target params	P_a	P_c	$\lvert\Sigma\rvert$
1	⟨ SG1, sICA, m1 ⟩	⟨ SG1, sICA, m2 ⟩	13/13	5/5	5
2	⟨ SG1, sICA, m1 ⟩	⟨ SG2, sICA, m2 ⟩	13/13	5/5	6
3	⟨ SG1, sICA, m1 ⟩	⟨ SG1, tPCA, m2 ⟩	10/13	5/5	6
4	⟨ SG1, sICA, m1 ⟩	⟨ SG2, tPCA, m2 ⟩	7/13	3/5	8
5	⟨ SG2, sICA, m1 ⟩	⟨ SG1, sICA, m2 ⟩	11/13	3/5	7
6	⟨ SG2, sICA, m1 ⟩	⟨ SG2, sICA, m2 ⟩	13/13	5/5	7
7	⟨ SG2, sICA, m1 ⟩	⟨ SG1, tPCA, m2 ⟩	10/13	5/5	6
8	⟨ SG2, sICA, m1 ⟩	⟨ SG2, tPCA, m2 ⟩	9/13	2/5	8
9	⟨ SG1, tPCA, m1 ⟩	⟨ SG1, sICA, m2 ⟩	7/13	5/5	4
10	⟨ SG1, tPCA, m1 ⟩	⟨ SG2, sICA, m2 ⟩	8/13	5/5	6
11	⟨ SG1, tPCA, m1 ⟩	⟨ SG1, tPCA, m2 ⟩	11/13	5/5	6
12	⟨ SG1, tPCA, m1 ⟩	⟨ SG2, tPCA, m2 ⟩	7/13	3/5	5
13	⟨ SG2, tPCA, m1 ⟩	⟨ SG1, sICA, m2 ⟩	7/13	3/5	5
14	⟨ SG2, tPCA, m1 ⟩	⟨ SG2, sICA, m2 ⟩	9/13	5/5	6
15	⟨ SG2, tPCA, m1 ⟩	⟨ SG1, tPCA, m2 ⟩	10/13	3/5	8
16	⟨ SG2, tPCA, m1 ⟩	⟨ SG2, tPCA, m2 ⟩	8/13	3/5	8

5 Conclusion

In this paper, we have presented a data mining approach to challenges in the matching and integration of heterogeneous datasets. In particular, we have proposed solutions to two problems that arise in combining information from different results of scientific research. The first problem, *attribute matching*, involves discovery of correspondences among distinct numeric-typed summary features ("attributes") that are used to characterize datasets that have been collected and analyzed in different research labs. The second problem, *cluster matching*, involves discovery of matchings between patterns across datasets.

We have treated both of these problems together as an multi-objective optimization problem. We developed a segmented statistics characterization to represent numeric-typed attributes and adapted the density profile to represent clusters. Based on these representations, we proposed objective functions that best define the criteria for selecting matching decisions. A multi-objective simulated annealing algorithm was described to find the optimal decision. The utility of this approach was demonstrated in a series of experiments using synthetic and realistic datasets that were designed to simulate heterogeneous data from different sources.

Acknowledgement. This work is supported by the NIH/NIBIB with Grant No. R01EB007684.

References

1. Bae, E., Bailey, J., Dong, G.: A Clustering Comparison Measure Using Density Profiles and Its Application to The Discovery tf Alternate Clusterings. Data Min. Knowl. Discov. 21, 427–471 (2010), http://dx.doi.org/10.1007/s10618-009-0164-z
2. Dhamankar, R., Lee, Y., Doan, A., Halevy, A., Domingos, P.: iMAP: Discovering Complex Semantic Matches between Database Schemas. In: Proceedings of the 2004 ACM SIGMOD International Conference on Management of Data. ACM Press (2004)
3. Dien, J.: The ERP PCA Toolkit: An Open Source Program for Advanced Statistical Analysis of Event-Related Potential Data. Journal of Neuroscience Methods 187(1), 138–145 (2010), http://www.sciencedirect.com/science/article/B6T04-4Y0KWB2-4/2/3c0e7b36b475b8d0e9a72c7b868a7dcd
4. Doan, A., Domingos, P., Levy, A.Y.: Learning Source Description for Data Integration. In: WebDB (Informal Proceedings), pp. 81–86 (2000), http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.42.9378
5. Fred, A.L., Jain, A.K.: Robust Data Clustering. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, p. 128 (2003)
6. Frishkoff, G.A., Frank, R.M., Rong, J., Dou, D., Dien, J., Halderman, L.K.: A Framework to Support Automated Classification and Labeling of Brain Electromagnetic Patterns. Computational Intelligence and Neuroscience (CIN), Special Issue, EEG/MEG Analysis and Signal Processing 7(3), 1–13 (2007)

7. Hamers, L., Hemeryck, Y., Herweyers, G., Janssen, M., Keters, H., Rousseau, R., Vanhoutte, A.: Similarity Measures In Scientometric Research: The Jaccard Index Versus Salton's Cosine Formula. Inf. Process. Manage. 25, 315–318 (1989), http://portal.acm.org/citation.cfm?id=67223.67231
8. Kuhn, H.W.: The Hungarian Method for The Assignment Problem. Naval Research Logistic Quarterly 2, 83–97 (1955)
9. Larson, J.A., Navathe, S.B., Elmasri, R.: A Theory of Attributed Equivalence in Databases with Application to Schema Integration. IEEE Trans. Softw. Eng. 15, 449–463 (1989), http://portal.acm.org/citation.cfm?id=63379.63387
10. Li, W.S., Clifton, C.: Semint: A Tool for Identifying Attribute Correspondences in Heterogeneous Databases Using Neural Networks (2000)
11. Liu, H., Frishkoff, G., Frank, R., Dou, D.: Ontology-Based Mining of Brainwaves: A Sequence Similarity Technique for Mapping Alternative Features in Event-Related Potentials (ERP) Data. In: Zaki, M.J., Yu, J.X., Ravindran, B., Pudi, V. (eds.) PAKDD 2010. LNCS, vol. 6119, pp. 43–54. Springer, Heidelberg (2010)
12. Rahm, E., Bernstein, P.A.: A Survey of Approaches to Automatic Schema Matching. VLDB Journal 10 (2001)
13. Rand, W.M.: Objective Criteria for the Evaluation of Clustering Methods. Journal of the American Statistical Association 66(336), 846–850 (1971), http://dx.doi.org/10.2307/2284239
14. Sheth, A.P., Larson, J.A., Cornelio, A., Navathe, S.B.: A Tool for Integrating Conceptual Schemas and User Views. In: Proceedings of the Fourth International Conference on Data Engineering, pp. 176–183. IEEE Computer Society, Washington, DC, USA (1988), http://portal.acm.org/citation.cfm?id=645473.653395
15. Suman, B.: Simulated annealing based multiobjective algorithm and their application for system reliability. Engin. Optim., 391–416 (2003)
16. Suman, B., Kumar, P.: A survey of simulated annealing as a tool for single!'/b?' and multiobjective optimization. Journal of the Operational Research Society 57, 1143–1160 (2006)
17. Zitzler, E., Thiele, L.: Multiobjective Optimization Using Evolutionary Algorithms - A Comparative Case Study, pp. 292–301. Springer, Heidelberg (1998)